WHAT IS CLAIMED IS:

1. A method of defining a simulated interactive Web page, the method comprising:

displaying on a computer display a programming area comprising one or more HTML user interface components for the simulated interactive Web page;

displaying on the computer display an area for primitives; and

enabling a user to draw a graphical coupling from a selected HTML user interface component to a selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the graphical coupling.

2. The method as defined in Claim 1, wherein the primitive is represented as a graphical symbol.

3. The method as defined in Claim 1, wherein the graphical coupling is a line.

4. The method as defined in Claim 1, wherein enabling a user to draw a graphical coupling further comprises:

monitoring a position of a cursor on the computer display, where the position of the cursor is manipulated by the user;

detecting that the user has selected the selected HTML user interface component from the interactive simulation area;

detecting that the cursor is guided from the selected HTML user interface component to the selected primitive; and

drawing the graphical coupling from the selected HTML user interface component to the selected primitive.

5. The method as defined in Claim 1, wherein the graphical coupling corresponds to data flow.

6. The method as defined in Claim 1, wherein the graphical coupling corresponds to determining which page is to be processed.

7. The method as defined in Claim 1, wherein the graphical coupling corresponds to determining which decision is to be processed.

8. The method as defined in Claim 1, wherein the selected primitive corresponds to an HTML user interface component.

9. The method as defined in Claim 1, wherein the selected primitive corresponds to a decision.

10. The method as defined in Claim 1, wherein the selected primitive corresponds to a reference.

11. The method as defined in Claim 1, wherein the selected primitive corresponds to a constant.

12. The method as defined in Claim 1, wherein the selected primitive corresponds to an action.

13. The method as defined in Claim 14, further comprising determining a graphical coupling type based at least partly on a type of the selected HTML user interface component and a type of the selected primitive.

14. The method as defined in Claim 1, wherein the selected primitive corresponds to at least one of how data is to be transformed, derived, stored, retrieved, and sorted.

15. The method as defined in Claim 1, further comprising enabling a user to draw a second graphical coupling from the selected primitive to a second selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the second graphical coupling.

16. The method as defined in Claim 15, wherein the graphical coupling corresponds to data flow.

17. The method as defined in Claim 15, wherein the graphical coupling corresponds to determining which page is to be processed.

18. The method as defined in Claim 15, wherein the graphical coupling corresponds to determining which decision is to be processed.

19. The method as defined in Claim 15, wherein the second selected primitive corresponds to an HTML user interface component.

20. The method as defined in Claim 15, wherein the second selected primitive corresponds to a decision.

21.     The method as defined in Claim 15, wherein the second selected primitive corresponds to a reference.

22.     The method as defined in Claim 15, wherein the second selected primitive corresponds to a constant.

23.     The method as defined in Claim 15, wherein the second selected primitive corresponds to an action.

24.     The method as defined in Claim 1, further comprising enabling a user to draw a second graphical coupling from a primitive representing a data flow label component to a second selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the second graphical coupling.

25.     The method as defined in Claim 1, wherein the selected primitive corresponds to another page and the selected HTML user interface component corresponds to a hyperlink, the method further comprising:

        enabling a user to remove the graphical coupling between the selected HTML user interface component and the selected primitive;

        determining that no other graphical couplings emanate from the selected HTML user interface component; and

        converting the hyperlink corresponding to the selected HTML user interface component to text based at least in part on the determination that no other lines emanate from the selected HTML user interface component without further action by the user.

26.     The method as defined in Claim 1, further comprising:

        enabling the user to create a subset of components that are related via a master element, where the subset includes HTML user interface components, primitives, and graphical couplings; and

        enabling the user to add a reference to the subset on the simulated interactive Web page as an HTML user interface component, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the subset corresponding to the reference.

27.     The method as defined in Claim 26, further comprising:

determining that the subset of components has been deleted; and

removing all references to the subset of components at least in part on the determination that the subset of components has been deleted without further action by the user.

28. The method as defined in Claim 26, wherein a graphical coupling between an HTML user interface component and a primitive corresponds to data flow.

29. The method as defined in Claim 26, wherein a graphical coupling between an HTML user interface component and a primitive corresponds to determining which page is to be processed.

30. The method as defined in Claim 26, wherein the graphical coupling between an HTML user interface component and a primitive corresponds to determining which decision is to be processed.

31. An interpreted interactive representation modeling apparatus that is executable in an interactive graphical user interface, the apparatus comprising:

a user interface component displayed in the interactive graphical user interface, where the user interface component includes at least two visible branches, where the visible branches are visible at least during configuration of the instruction for the interpreted interactive representation modeling language;

a Boolean condition associated with at least a first branch and a second branch of the at least two visible branches, where the first branch is associated with a first state of the Boolean condition, and where the second branch is associated with a second state of the Boolean condition, where the association of the Boolean condition is visibly displayed in the interactive graphical user interface at least during configuration of the instruction;

an interactive component responsive to user interaction during execution of the instruction, where the interactive component monitors data flow inputs to the interactive graphical user interface for a selection of a state of the Boolean condition;

at least a first executable instruction associated with the first branch, where the first executable instruction is activated upon the detection of a selection of the first

state of the Boolean condition such that the first executable instruction is conditionally executed; and

at least a second executable instruction associated with the second branch, where the second executable instruction is activated upon the detection of a selection of the second state of the Boolean condition such that the second executable instruction is conditionally executed.

32. The apparatus as defined in Claim 31, wherein the interactive graphical user interface is rendered by a browser.

33. The apparatus as defined in Claim 31, wherein the first state associated with the first branch and the second state associated with the second branch are automatically determined upon configuration of the instruction.

34. The apparatus as defined in Claim 31, wherein the first executable instruction and the second executable instruction are programmed via interaction with the interactive graphical user interface.

35. The apparatus as defined in Claim 31, wherein the first executable instruction and the second executable instruction comprise hypertext markup language (HTML) statements.

36. The apparatus as defined in Claim 31, wherein the first state is false and the second state is true.

37. The apparatus as defined in Claim 31, wherein the apparatus is further configured to execute a predetermined one of at least the first executable instruction or the second executable instruction by default when the interactive component responsive to user interaction does not detect a data flow input corresponding to a selection of a state of the Boolean condition.

38. The apparatus as defined in Claim 31, wherein the apparatus is further configured to enter an integer mode upon the adding of a third branch, the first branch is associated with a first integer value of an integer-mode condition, the second branch is associated with a second integer value of the integer-mode condition, the third branch is associated with a third integer value of the integer-mode condition, and wherein the association of the integer-mode condition is visibly displayed.

39.     An interpreted interactive representation modeling apparatus, the apparatus comprising:

a user interface component displayed in an interactive graphical user interface, where the user interface component includes at least three visible branches, where the visible branches are visible at least during configuration of the instruction for the interpreted interactive representation modeling language;

an integer-mode condition associated with at least a first branch, a second branch, and a third branch of the at least three visible branches, wherein:

the first branch is associated with a first integer value of the integer-mode condition;

the second branch is associated with a second integer value of the integer-mode condition;

the third branch is associated with a third integer value of the integer-mode condition;

wherein the association of the integer-mode condition is visibly displayed at least during configuration of the instruction;

an interactive component responsive to user interaction during execution of the instruction, where the interactive component monitors for a selection of a value for the integer-mode condition;

at least a first executable instruction associated with the first branch, where the first executable instruction is activated upon the detection of a value corresponding to the first integer value of the integer-mode condition;

at least a second executable instruction associated with the second branch, where the second executable instruction is activated upon the detection of a value corresponding to the second integer value of the integer-mode condition; and

at least a third executable instruction associated with the third branch, where the third executable instruction is activated upon the detection of a value corresponding to the third integer value of the integer-mode condition.

40.     The apparatus as defined in Claim 39, wherein the interactive graphical user interface is displayed via a browser.

41. The apparatus as defined in Claim 39, wherein the first executable instruction, the second executable instruction, and the third executable instruction are programmed via interaction with the interactive graphical user interface.

42. The apparatus as defined in Claim 39, wherein the first executable instruction, the second executable instruction, and the third executable instruction comprise hypertext markup language (HTML) statements.

43. The apparatus as defined in Claim 39, wherein the first value corresponds to 0, the second value corresponds to 1, and the third value corresponds to 2.

44. The apparatus as defined in Claim 39, wherein the apparatus is further configured to execute a predetermined one of at least the first executable instruction, the second executable instruction, or the third executable instruction by default when the interactive component responsive to user interaction does not detect a data flow input corresponding to a value for the integer-mode condition.

45. A method of interacting with a user to define a behavior of a portion of an interactive presentation, where the interactive presentation corresponds to a simulation model, where the interactive presentation and the user interact via an interactive graphical user interface, the method comprising:

monitoring a manipulation of a cursor by a pointing device;

monitoring a graphical dragging and dropping of a first graphical symbol to a first area in the interactive graphical user interface such that a user interface component appears in the first area, where the user interface component corresponds to a conditionally-executed instruction;

receiving a name for the user interface component;

receiving a name and a text description for a requirement for the user interface component;

identifiably storing the name of the user interface component, the name of the requirement, and the text description of the requirement with the user interface component;

monitoring a dropping of a second graphical symbol into the first graphical symbol in the pane of the interactive graphical user interface, and at least partly in

response to the dropping of the second graphical symbol, adding a first branch to the first graphical symbol, where the first branch corresponds to a first code that is executed upon selection of the first branch during run time;

monitoring a dropping of a third graphical symbol into the first graphical symbol in the pane of the interactive graphical user interface, and at least partly in response to the dropping of the third graphical symbol, adding a second branch to the first graphical symbol, where the second branch corresponds to a second code that is executed upon selection of the second branch during run time; and

receiving a data flow input control for the first graphical symbol, where the data flow input is associated with the first graphical symbol such that when the interactive simulation of the simulation model runs in the interactive graphical user interface, a combination of the data flow input control and a data flow input received during run-time control the branching of the first graphical symbol to the first branch and to the second branch.

46.     The method as defined in Claim 45, wherein the first code and the second code correspond to hypertext markup language (HTML) code statements.

47.     The method as defined in Claim 45, where the data flow input control corresponds to a Boolean expression.

48.     The method as defined in Claim 45, further comprising:

monitoring a dropping of a fourth graphical symbol into the first graphical symbol in the first area of the interactive graphical user interface, and at least partly in response to the dropping of the fourth graphical symbol, adding a third branch to the first graphical symbol, where the third branch corresponds to a third code that is executed upon selection of the third branch; and

where the data flow input control for the first graphical symbol corresponds to an integer-mode control, and where the combination of the data flow input control and a data flow input received during runtime control the branching of the first graphical symbol to the first branch, to the second branch, or to the third branch.

49. A method of automatically configuring at least a portion of a behavior for an instruction for a requirements validation computer program, the method comprising:

providing a user interface component in an interactive graphical user interface;

monitoring conditional branches added to the user interface component, where the conditional branches control program flow for the requirements validation computer program;

automatically associating the conditional branches with a Boolean state when there are two conditional branches associated with the user interface component; and

automatically associating the conditional branches with an integer-mode when there are at least three conditional branches associated with the user interface component.

50. The method as defined in Claim 49, wherein the conditional branches are further associated with corresponding instructions that are executed upon the selection of the corresponding branch.

51. The method as defined in Claim 50, wherein the instructions associated with the conditional branches comprise hypertext markup language (HTML) statements.

52. A computer system that is configured to define a simulated interactive Web page, the computer system comprising:

a component configured to display on a computer display a programming area comprising one or more HTML user interface components for the simulated interactive Web page;

a component configured to display on the computer display an area for primitives; and

a component configured to enable a user to draw a graphical coupling from a selected HTML user interface component to a selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the graphical coupling.

53. The computer system as defined in Claim 52, further comprising a component configured to enablea user to draw a second graphical coupling from the selected primitive to

a second selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the second graphical coupling.

54.    The computer system as defined in Claim 52, further comprising a component configured to enable a user to draw a second graphical coupling from a primitive representing a data flow label component to a second selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the second graphical coupling.

55.    The computer system as defined in Claim 52, wherein the selected primitive corresponds to another page and the selected HTML user interface component corresponds to a hyperlink, the computer system further comprising:

   a component configured to enable a user to remove the graphical coupling between the selected HTML user interface component and the selected primitive;

   a component configured to determine that no other graphical couplings emanate from the selected HTML user interface component; and

   a component configured to convert the hyperlink corresponding to the selected HTML user interface component to text based at least in part on the determination that no other lines emanate from the selected HTML user interface component without further action by the user.

56.    The computer system as defined in Claim 52, further comprising:

   a component configured to enable the user to create a subset of components that are related via a master element, where the subset includes HTML user interface components, primitives, and graphical couplings; and

   a component configured to enable the user to add a reference to the subset on the simulated interactive Web page as an HTML user interface component, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the subset corresponding to the reference.

57.    The computer system as defined in Claim 56, further comprising:

   a component configured to determine that the subset of components has been deleted; and

a component configured to remove all references to the subset of components at least in part on the determination that the subset of components has been deleted without further action by the user.

58.     A computer system that is configured to define a simulated interactive Web page, the computer system comprising:

a means for displaying on a computer display a programming area comprising one or more HTML user interface components for the simulated interactive Web page;

a means for displaying on the computer display an area for primitives; and

a means for enabling a user to draw a graphical coupling from a selected HTML user interface component to a selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the graphical coupling.

59.     The computer system as defined in Claim 58, further comprising a means for enabling a user to draw a second graphical coupling from the selected primitive to a second selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the second graphical coupling.

60.     A computer program embodied in a tangible medium for defining a simulated interactive Web page, the computer program comprising:

instructions configured to display on a computer display a programming area comprising one or more HTML user interface components for the simulated interactive Web page;

instructions configured to display on the computer display an area for primitives; and

instructions configured to enable a user to draw a graphical coupling from a selected HTML user interface component to a selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the graphical coupling.

61.     The computer program as defined in Claim 60, further comprising instructions configured to enable a user to draw a second graphical coupling from the selected primitive

to a second selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the second graphical coupling.

62.     The computer program as defined in Claim 60, further comprising instructions configured to enable a user to draw a second graphical coupling from a primitive representing a data flow label component to a second selected primitive, wherein execution of the simulated interactive Web page is based at least in part on an interpretation of the second graphical coupling.

63.     The computer program as defined in Claim 60, wherein the selected primitive corresponds to another page and the selected HTML user interface component corresponds to a hyperlink, the computer program further comprising:

instructions configured to enable a user to remove the graphical coupling between the selected HTML user interface component and the selected primitive;

instructions configured to determine that no other graphical couplings emanate from the selected HTML user interface component; and

instructions configured to convert the hyperlink corresponding to the selected HTML user interface component to text based at least in part on the determination that no other lines emanate from the selected HTML user interface component without further action by the user.